

# Data 583 Life Expectancy (WHO)

Justin Chan, Kenny Tong, Viji Rajagopalan

7 Mar, 2023

## Data 583 Life Expectancy - Exploratory Data Analysis (Life Expectancy Data)

### Introduction

The main goal of assessing the Life expectancy dataset is to understand the variables that impact life expectancy and also to know how well life expectancy is described by the selected variables. In this document, we take a detailed look at different aspects of the data like summary and apply statistical techniques to understand the underlying structures. Also, based on the assessments, we will come up with next steps to fine tune data and improve modeling.

### Data Exploration

#### Original Dataset Summary & Initial Data Screening

##### Purpose

- Let's take a snapshot of the original dataset and have a rough idea of its record

**Procedure** Take a look at the dataset summary.

```
summary(le)
```

```
##      Country      Year      Status      Life.expectancy
## Length:2938      Min.   :2000      Length:2938      Min.   :36.30
## Class :character  1st Qu.:2004      Class :character  1st Qu.:63.10
## Mode  :character  Median :2008      Mode  :character  Median :72.10
##                               Mean   :2008              Mean   :69.22
##                               3rd Qu.:2012              3rd Qu.:75.70
##                               Max.   :2015              Max.   :89.00
##                               NA's   :10
## Adult.Mortality infant.deaths      Alcohol      percentage.expenditure
## Min.   : 1.0      Min.   : 0.0      Min.   : 0.0100      Min.   : 0.000
## 1st Qu.: 74.0      1st Qu.: 0.0      1st Qu.: 0.8775      1st Qu.: 4.685
## Median :144.0      Median : 3.0      Median : 3.7550      Median : 64.913
## Mean   :164.8      Mean   : 30.3      Mean   : 4.6029      Mean   : 738.251
## 3rd Qu.:228.0      3rd Qu.: 22.0      3rd Qu.: 7.7025      3rd Qu.: 441.534
## Max.   :723.0      Max.   :1800.0      Max.   :17.8700      Max.   :19479.912
## NA's   :10              NA's   :194
## Hepatitis.B      Measles      BMI      under.five.deaths
## Min.   : 1.00      Min.   : 0.0      Min.   : 1.00      Min.   : 0.00
## 1st Qu.:77.00      1st Qu.: 0.0      1st Qu.:19.30      1st Qu.: 0.00
## Median :92.00      Median : 17.0      Median :43.50      Median : 4.00
```

```

## Mean :80.94 Mean : 2419.6 Mean :38.32 Mean : 42.04
## 3rd Qu.:97.00 3rd Qu.: 360.2 3rd Qu.:56.20 3rd Qu.: 28.00
## Max. :99.00 Max. :212183.0 Max. :87.30 Max. :2500.00
## NA's :553 NA's :34
## Polio Total.expenditure Diphtheria HIV.AIDS
## Min. : 3.00 Min. : 0.370 Min. : 2.00 Min. : 0.100
## 1st Qu.:78.00 1st Qu.: 4.260 1st Qu.:78.00 1st Qu.: 0.100
## Median :93.00 Median : 5.755 Median :93.00 Median : 0.100
## Mean :82.55 Mean : 5.938 Mean :82.32 Mean : 1.742
## 3rd Qu.:97.00 3rd Qu.: 7.492 3rd Qu.:97.00 3rd Qu.: 0.800
## Max. :99.00 Max. :17.600 Max. :99.00 Max. :50.600
## NA's :19 NA's :226 NA's :19
## GDP Population thinness..1.19.years
## Min. : 1.68 Min. :3.400e+01 Min. : 0.10
## 1st Qu.: 463.94 1st Qu.:1.958e+05 1st Qu.: 1.60
## Median : 1766.95 Median :1.387e+06 Median : 3.30
## Mean : 7483.16 Mean :1.275e+07 Mean : 4.84
## 3rd Qu.: 5910.81 3rd Qu.:7.420e+06 3rd Qu.: 7.20
## Max. :119172.74 Max. :1.294e+09 Max. :27.70
## NA's :448 NA's :652 NA's :34
## thinness.5.9.years Income.composition.of.resources Schooling
## Min. : 0.10 Min. :0.0000 Min. : 0.00
## 1st Qu.: 1.50 1st Qu.:0.4930 1st Qu.:10.10
## Median : 3.30 Median :0.6770 Median :12.30
## Mean : 4.87 Mean :0.6276 Mean :11.99
## 3rd Qu.: 7.20 3rd Qu.:0.7790 3rd Qu.:14.30
## Max. :28.60 Max. :0.9480 Max. :20.70
## NA's :34 NA's :167 NA's :163

```

Here is another overview that gives information on type of variable, sample data from each column below. This also gives number of records and columns which is 2938 and 22 respectively.

```

## 'data.frame': 2938 obs. of 22 variables:
## $ Country : chr "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...
## $ Year : int 2015 2014 2013 2012 2011 2010 2009 2008 2007 2006 ...
## $ Status : chr "Developing" "Developing" "Developing" "Developing" ...
## $ Life.expectancy : num 65 59.9 59.9 59.5 59.2 58.8 58.6 58.1 57.5 57.3 ...
## $ Adult.Mortality : int 263 271 268 272 275 279 281 287 295 295 ...
## $ infant.deaths : int 62 64 66 69 71 74 77 80 82 84 ...
## $ Alcohol : num 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.01 0.03 0.02 0.03 ...
## $ percentage.expenditure : num 71.3 73.5 73.2 78.2 7.1 ...
## $ Hepatitis.B : int 65 62 64 67 68 66 63 64 63 64 ...
## $ Measles : int 1154 492 430 2787 3013 1989 2861 1599 1141 1990 ...
## $ BMI : num 19.1 18.6 18.1 17.6 17.2 16.7 16.2 15.7 15.2 14.7 ...
## $ under.five.deaths : int 83 86 89 93 97 102 106 110 113 116 ...
## $ Polio : int 6 58 62 67 68 66 63 64 63 58 ...
## $ Total.expenditure : num 8.16 8.18 8.13 8.52 7.87 9.2 9.42 8.33 6.73 7.43 ...
## $ Diphtheria : int 65 62 64 67 68 66 63 64 63 58 ...
## $ HIV.AIDS : num 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 ...
## $ GDP : num 584.3 612.7 631.7 670 63.5 ...
## $ Population : num 33736494 327582 31731688 3696958 2978599 ...
## $ thinness..1.19.years : num 17.2 17.5 17.7 17.9 18.2 18.4 18.6 18.8 19 19.2 ...
## $ thinness.5.9.years : num 17.3 17.5 17.7 18 18.2 18.4 18.7 18.9 19.1 19.3 ...
## $ Income.composition.of.resources : num 0.479 0.476 0.47 0.463 0.454 0.448 0.434 0.433 0.415 0.405 ...
## $ Schooling : num 10.1 10 9.9 9.8 9.5 9.2 8.9 8.7 8.4 8.1 ...

```

## Conclusion/Key Findings :

- The records range is from Year 2000 to 2015
- Columns with NA : Life Expectancy, Adult Mortality, Alcohol, Hep B, BMI, Polio, Total exp, Dip, GDP, Population, thinness..1.19, thinness.5.9, Income.composition.of.resources, Schooling
- ‘Status’ Column is of the “character” data type, with values “Developing” and “Developed”. We will introduce a new column ‘Status.val’ to be the factor value of ‘Status’ for better analysis..
- ‘Percentage Expenditure’ has a mean value of 738.2512955. Spending on health is more than the GDP per capita (!?). Look into the column definition : Expenditure on health as a percentage of Gross Domestic Product per capita(%). The data of such magnitude simply does not quite make sense. Cross check with other references (e.g. the World Bank <https://data.worldbank.org/indicator/SH.XPD.CHEX.GD.ZS>). OK, let’s conclude that we have hesitation about the reliability/interpretation of the value of this column, and this column would be dropped for the rest of this analysis.

## Null Value Analysis and Handling

### Purpose

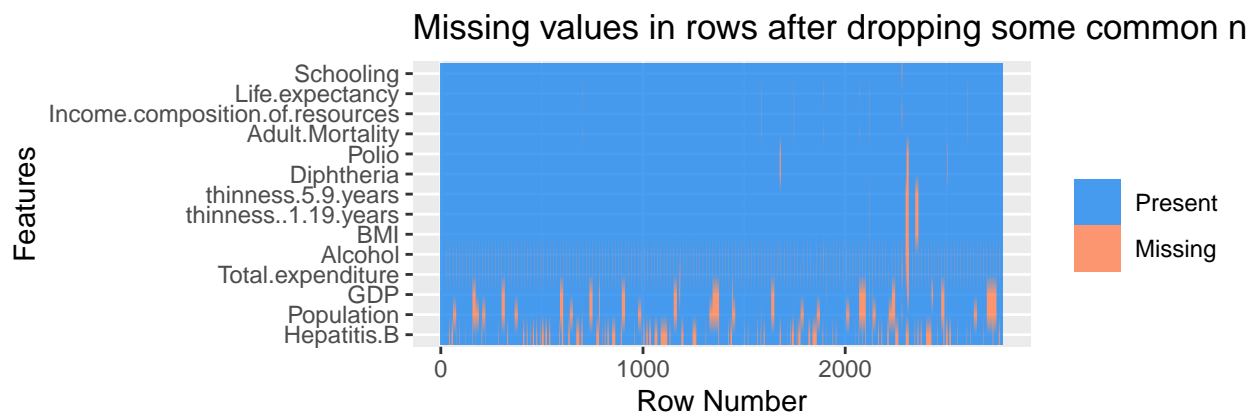
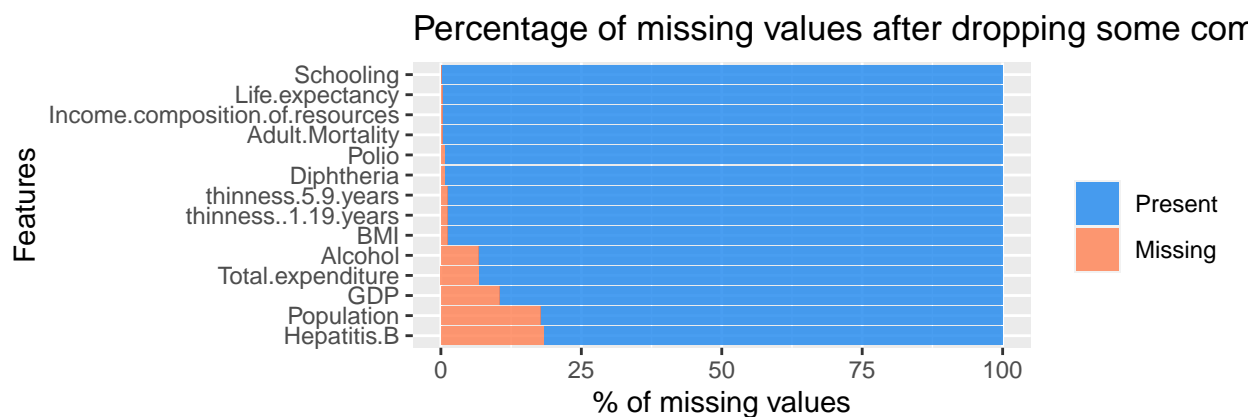
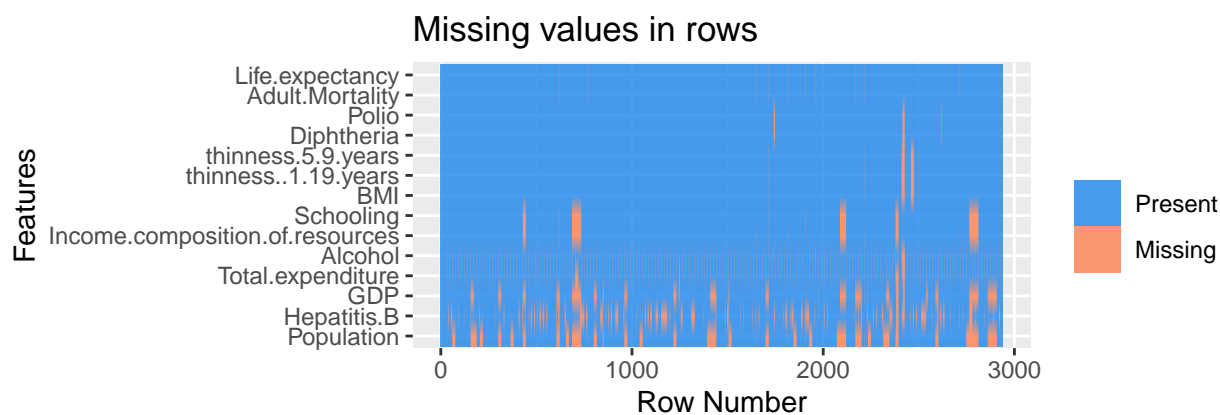
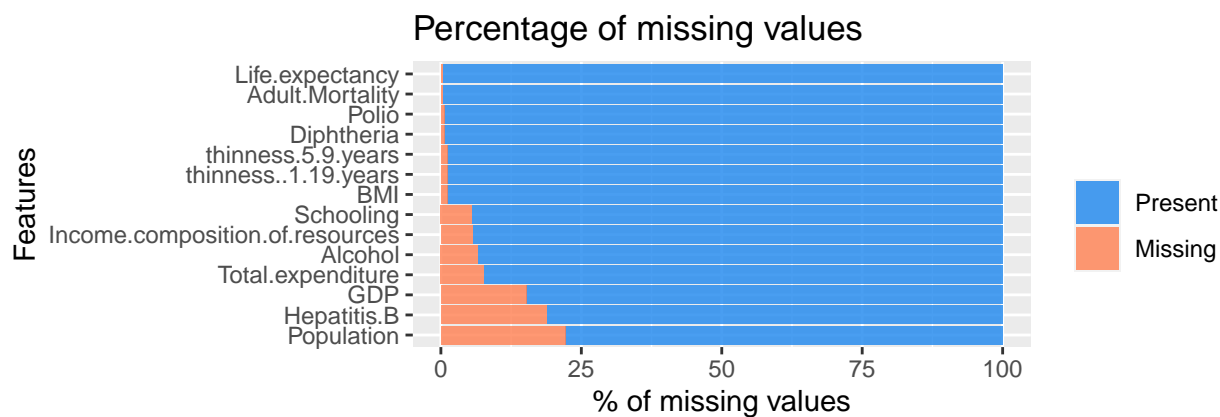
- Investigate the and determine how to handle the null value in the data set. Missing values could have a large affect to the overall quality of the static models and machine learning models and need to be clean before using it in our training model.

**Procedure** Lets investigate how many missing values within our features.

A brief check indicates that there are total of 2563 missing value within our dataset, we could visualize the missing data to identify patterns or cluster of missing values within our data to determine the cause of the missing data and whether it is random or systematic and to highlight potential biases that may exist in our data set. Visualizing the missing value also allow to understand the extend of the missing data and determine appropriate strategies for imputing missing value. Refer figure in next page, there are 2938 no. of rows in the dataset. According to our Visualization, there seems to be a correlation in the appearance in missing data in our original data’s feature “population”, “gdp” , “income.composition.of.resources” and “schooling”. We would deal with this correlation in missing data by removing the the record that have missing value in all of the listed variables. For the other values, we would set the na to the respective column mean for the subsequent analysis.

## Conclusion/Key Findings :

- na values have been analyzed
- Data imputation have been performed as far as possible in order to prepare for the subsequent data analysis.



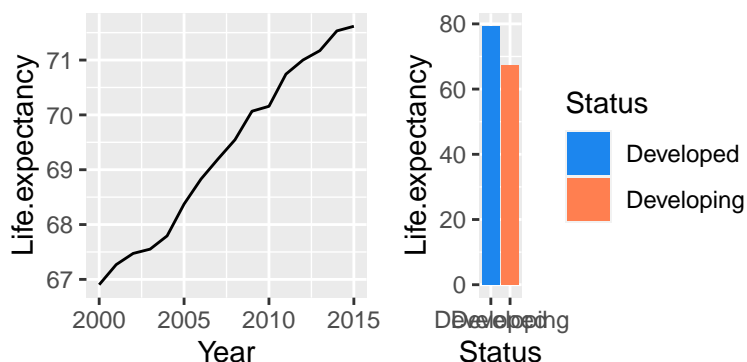
## Statistical Analysis

**Purpose** Do some visualization and initial statistical model assessments to explore and identify the general data pattern, trends and clusters, etc.

### Procedure

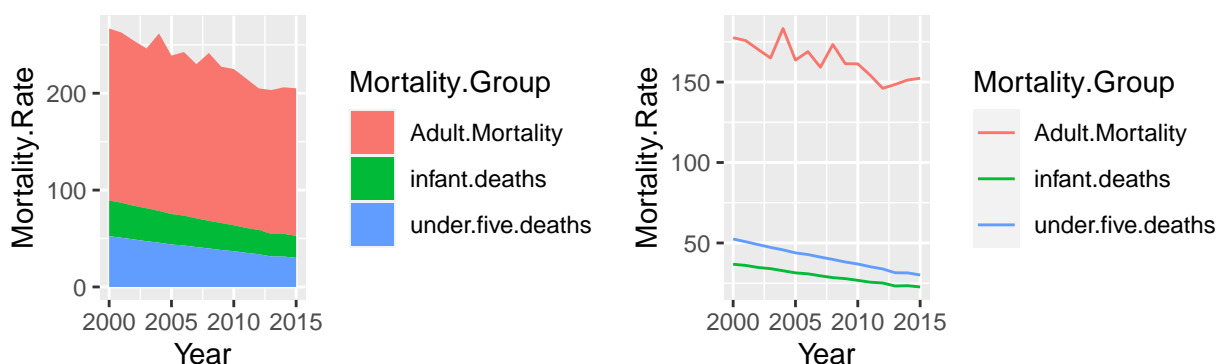
#### General Life Expectancy

As we are interested in Life Expectancy as our response variable, we first start looking at the distribution of the variable and general trend.



#### Conclusion/Key Findings :

- The general life expectancy has been steadily increasing duration the year
- Average Life expectancy increase from about 67 to 71.5 in 15 years
- Life expectancy of Developed countries are significantly higher than that of Developing countries.



#### Findings :

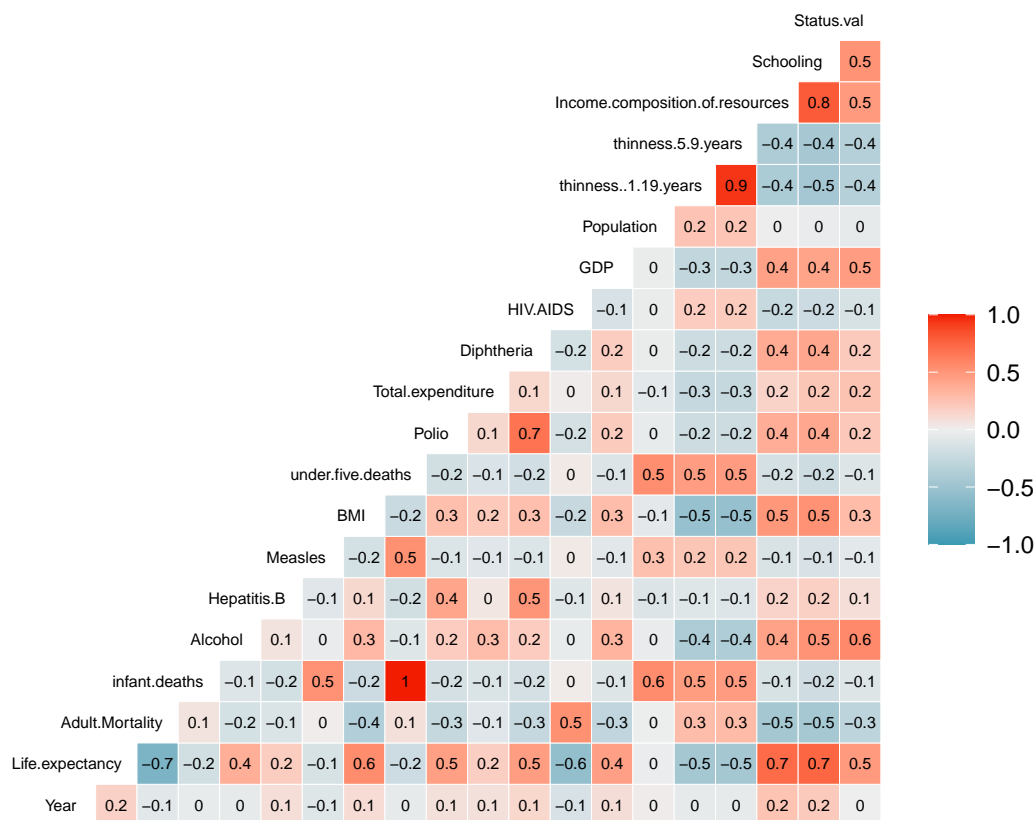
- The mortality rate of all three age groups are generally decreasing as a whole
- The mortality rate of the adult group, however, have fluctuation within the period

For rest of the analysis, we drop the country column so that we can understand general trends in life expectancy independent of countries. Let's visualize the distribution of response variable using below histogram.

Findings : - As we see, the response variable Life Expectancy is normally distributed and so our first try is to see if MLR is able to predict well. - Also, using this model and running a BIC on it, we can understand the columns that are important.

## Predictor Space

We now turn our focus to look at the different predictor variables. Following shows the correlation of different variables and their spread in the dataset.



Summary of correlation for  $>0.7$  or  $<-0.7$ : infant.deaths and under.five.deaths is nearly 100% correlated with each other and Polio is highly correlated with Diphtheria, Income composition of resources is highly correlated with Schooling. Both thinness variables are highly correlated. Other variables are low to moderately correlated. The response variable Life expectancy is highly correlated with Income composition, Schooling and adult mortality variables.

### Initial Modelling and Variable Importance:

As response variable Life.Expectancy is approximately normally distributed, first step is to try lm model for this data and also run BIC to get the variable selection from the dataset.

The dataset has 20+ predictors and based on correlation plot there are correlation between the variables, BIC would help to eliminate some of the predictors that are conveying same signal as others and also explains less variability in life expectancy.

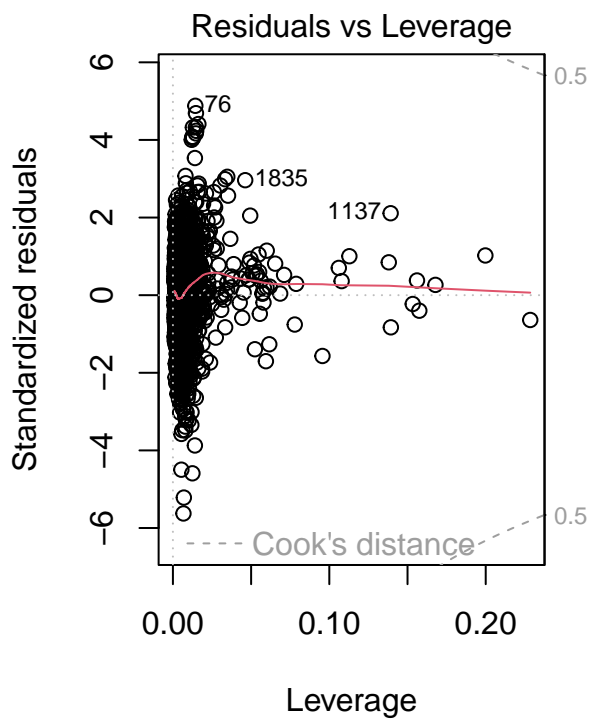
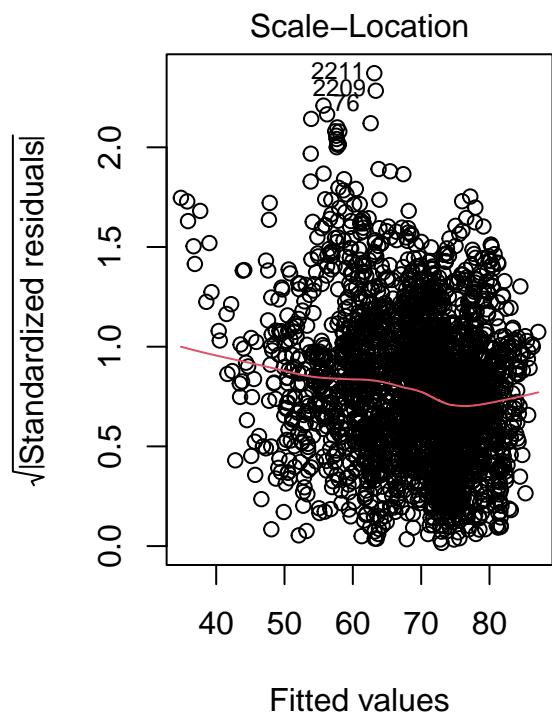
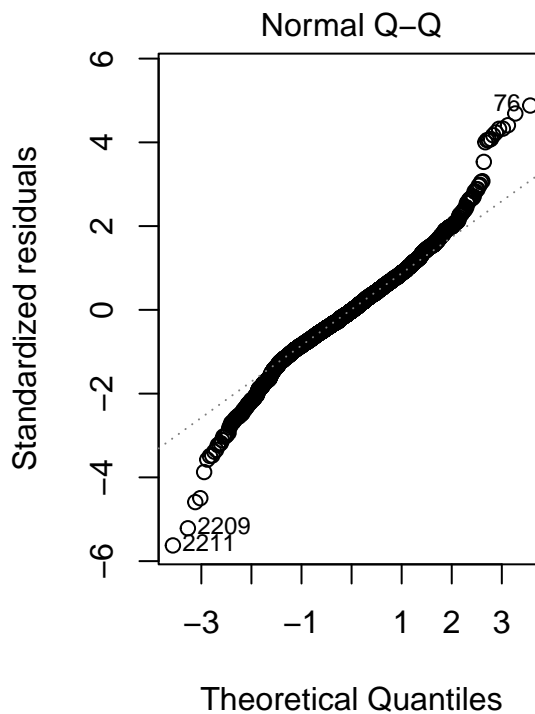
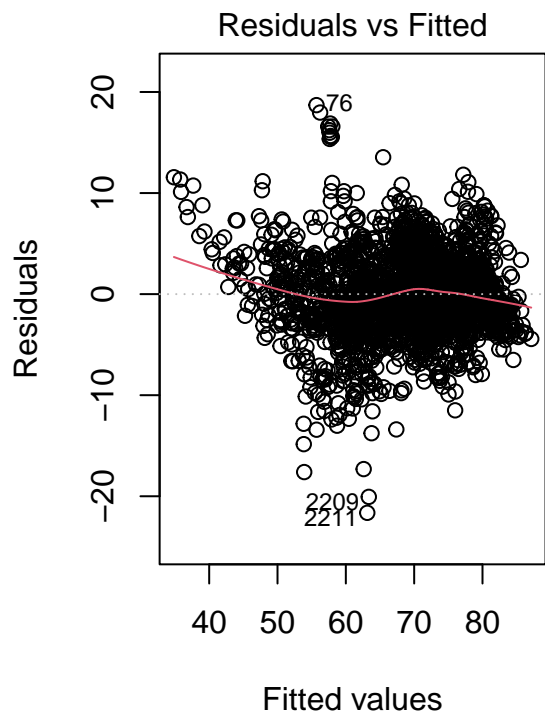
### Full model summary and diagnostics

```
##
## Call:
## lm(formula = Life.expectancy ~ ., data = df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
```

```

## -21.6365 -2.1904 -0.0813 2.2726 18.6972
##
## Coefficients: (1 not defined because of singularities)
##
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.363e+02 3.413e+01 3.992 6.71e-05 ***
## Year -4.046e-02 1.707e-02 -2.369 0.017888 *
## StatusDeveloping -1.340e+00 2.679e-01 -5.002 6.02e-07 ***
## Adult.Mortality -1.674e-02 7.966e-04 -21.011 < 2e-16 ***
## infant.deaths 9.138e-02 8.101e-03 11.280 < 2e-16 ***
## Alcohol -2.433e-02 2.584e-02 -0.941 0.346621
## Hepatitis.B -1.387e-02 3.833e-03 -3.617 0.000303 ***
## Measles -9.298e-06 8.300e-06 -1.120 0.262740
## BMI 3.946e-02 4.887e-03 8.074 1.01e-15 ***
## under.five.deaths -6.807e-02 5.944e-03 -11.453 < 2e-16 ***
## Polio 2.489e-02 4.482e-03 5.554 3.06e-08 ***
## Total.expenditure 5.942e-02 3.421e-02 1.737 0.082527 .
## Diphtheria 3.343e-02 4.749e-03 7.040 2.41e-12 ***
## HIV.AIDS -4.878e-01 1.705e-02 -28.609 < 2e-16 ***
## GDP 5.272e-05 6.444e-06 8.182 4.23e-16 ***
## Population -1.033e-09 1.618e-09 -0.639 0.523086
## thinness..1.19.years -7.134e-02 4.860e-02 -1.468 0.142241
## thinness.5.9.years 6.001e-03 4.793e-02 0.125 0.900374
## Income.composition.of.resources 6.736e+00 6.152e-01 10.949 < 2e-16 ***
## Schooling 7.798e-01 4.083e-02 19.097 < 2e-16 ***
## Status.val NA NA NA NA
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.86 on 2758 degrees of freedom
## Multiple R-squared: 0.831, Adjusted R-squared: 0.8299
## F-statistic: 713.8 on 19 and 2758 DF, p-value: < 2.2e-16

```



Residuals related plots: The residuals versus fits plot would provide us with information on the residual against the fitted values in regression analysis. This could be used to identify the patterns in the residuals that may indicate



the model is not capturing the relationship between our predictor and the outcome variable, therefore, allowing us to detect any non-linearity, unequal error variances and outliers. In general, we would want to see our residual randomly scattered around 0 since this indicates that the model assumption is met and is a good fit for the data. However, from the above-plotted residuals versus fits plot, we could see there is a curvature shape to our residuals and there is a presence of outliers and high leverage points on the left-hand side of the residuals versus fits the plot, This could be problematic since outliers and leverage points could have a significant impact on the regression coefficient. And the curved shape indicates that our model may be misspecified and further investigation is needed. Also, same is indicated by the scale-location plot with standardized residuals  $> 1.5$ . Based on leverage plot, we do not see the need to remove any outliers in this initial assessment.

QQ plot: QQ-plot (Quantile-quantile plot) allow us to investigate the univariate normality of the dataset. If the points on the QQ-plot fall approximately along a straight line, it suggests that the sample comes from a population with similar distribution to the theoretical distribution that we are comparing to. From the QQ-plot that we have plotted above, the point deviates from a straight line on both ends and indicates there is a heavy tail.

We make a note of the structures in the initial assessment and acknowledge they could impact the MLR performance. During next stage of the project, we plan to address the issues identified in diagnostics during final modeling.

**Variable selection methods** Following is the summary from a couple of common methods used in variable section of models: BIC(Bayesian information criterion, backward selection) and VIF (Variable Inflation Factor).

Initially, all 20 variables were used in our model and achieved an AIC score of 7642.14. After performing BIC backward step model selection method, The BIC backward step model selection method has reduced our model's independent variable to 13 and achieved a lower AIC score of 7604.34. Since a lower AIC score signifies the regression is a better fit to the data, meaning that after removing some irrelevant variable in our data set, the simple model is still able to explain the data well well and has improved the fitting from the initial model. Also the final reduced model has an Adjusted R-squared score of 0.8296 compared to the original model's 0.8299 isn't much of a drop in the Adjusted R-squared score meaning our BIC reduced model was still able to have the same amount of variability.

Now going back to variable selection, this time we will be using VIF(Variance Inflation Factor) to investigate whether it is possible to come up with a better model from BIC reduced model by eliminating some highly correlated variables in the data.

VIF(Variance Inflation Factor) is a variable selection method that is used to identify and eliminate highly correlated variables in a regression model. If the VIF value for a variable is high, it indicates that the variable is highly correlated with another predictor within the model. From the above output, we could see that variables "infant.deaths" and "under.five.deaths" are highly correlated. Therefore we will have to remove either "infant.deaths" or "under.five.deaths" to resolve the multicollinearity within our data inorder to improve our model accuracy and interpretability. We acknowledge this may be necessary for next level of tuning the model.

After doing variable reduction and selecting only variables that are recommended by BIC (Status + Adult.Mortality + infant.deaths + Hepatitis.B + BMI + under.five.deaths + Polio + Diphtheria + HIV.AIDS + GDP + thinness.1.19.years + Income.composition.of.resources + Schooling), we move on checking for any clustering effects in data.

```
library(mclust)
clus1 <- Mclust(df1)
summary(clus1)

## -----
## Gaussian finite mixture model fitted by EM algorithm
## -----
##
## Mclust VEV (ellipsoidal, equal shape) model with 7 components:
##
## log-likelihood    n  df      BIC      ICL
##      -115852.1 2778 761 -237738.5 -237871.7
##
```

```
## Clustering table:
##   1   2   3   4   5   6   7
## 703 372 294 330 246 533 300
```

**Clustering** Looks like there are some clusters in the data, first understanding is it is because of variable “Status”’s developed vs developing. Acknowledging this information which may be helpful in future phases of model building and fine tuning. For example, if MLR would be the final model, building interaction with cluster variable and rest of data would further improve model performance.

Using our reduced model, we feed that data to Linear Model and achieve an Adjusted R-squared of 0.8296 and all variables have a p-value of less the 0.05, meaning that all of our independent variables within the model is statically significant to our dependent variable, In other words, there is strong evidence against the null hypothesis, suggesting that the observed relation between other dependent variable and independent variable is significant and real, not just due to random variation or chance.

## Conclusion/Key Findings

Summarized below are some key findings from EDA.

- Response variable is looking to be normally distributed and initial model score is ~82% which means this model is able to explain 82% of variation in Life expectancy. Its possible to use multiple linear regression for this data. From the diagnostic plots it may be seen that there is skewness in the data.
- There are some variables that suffer multi collinearity (from VIF) scores.
- Not all predictors are necessary to describe response variable. Model selection will be helpful.

Due to the spread of data (clustering, non-linearity of predictors w response variable, skewness in data), it is necessary to explore other models specifically non-parametric regression models.

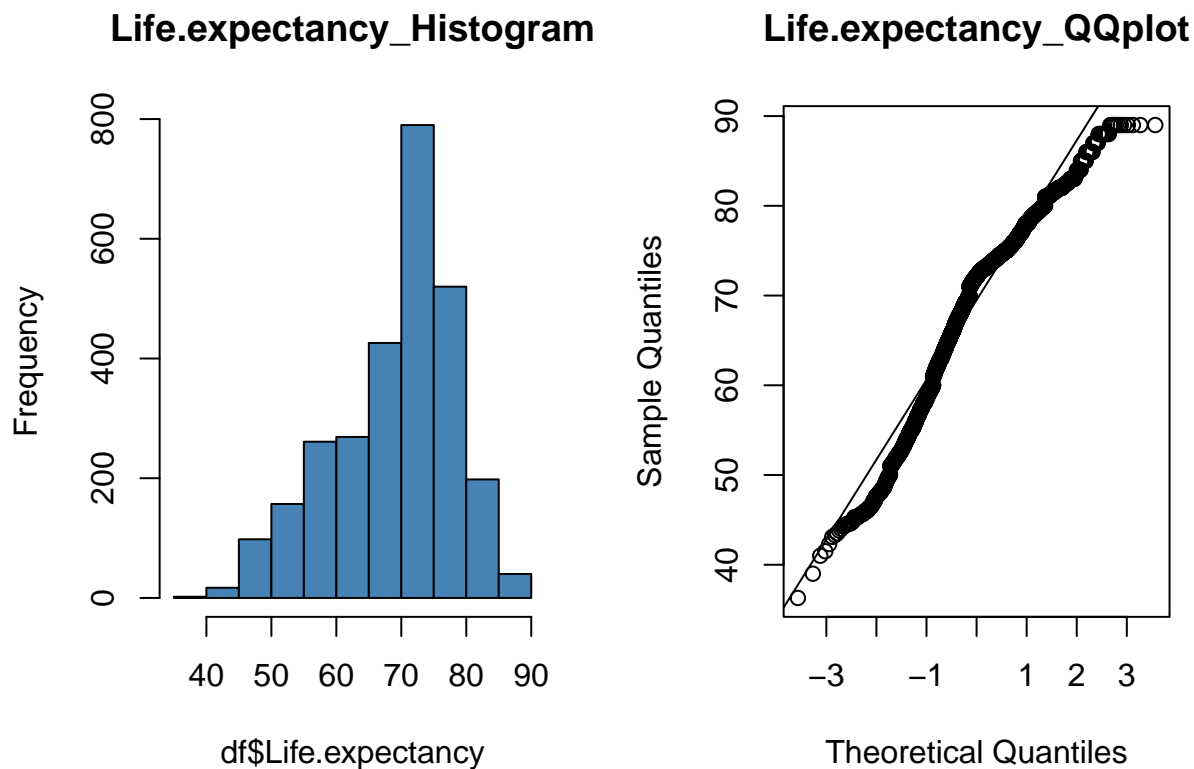
## Questions and Next Steps

1. Does the variables selected using BIC and linear model able to explain Life Expectancy adequately? A hypotheses test is required for this.
2. Is the response variable normally distributed? Shapiro-Wilk test for normality will need to be conducted for this.
3. Is Multi linear regression the best model or go with other non parametric models? From initial feedback, there are hypotheses tests available to validate this.

Need to explore further on these questions from proposal stage and conclude.

4. Understanding impact of individually controlled factors - The dataset has all predicting variables divided into 4 groups: Immunization related factors, Mortality factors, Economical factors and Social factors. Some of these factors are controllable by individuals like immunization, alcohol etc. Some of these factors are noncontrollable and macro elements like GDP. If an individual within a country want to improve life expectancy, how much is controllable/can be influenced personally? What proportion of variation in life expectancy can be explained by these variables? For example, What is the effect of “Alcohol/BMI” on the life expectancy?
5. Understanding impact of Government/Public controlled factors - From Government perspective, how are the preventive measures influencing life expectancy? What proportion of variation in life expectancy can be explained by these variables? For example, Does Higher health expenditure (column H) on Health improve life expectancy?
6. normal distribution test for our y variable

```
#Histogram & QQPlot
par(mfrow=c(1,2))
hist(df$Life.expectancy, col='steelblue', main='Life.expectancy_Histogram')
#not really a good "bell-shape"
qqnorm(df$Life.expectancy, main='Life.expectancy_QQplot')
#most of the data is not fall along a straight diagonal line
qqline(df$Life.expectancy)
```



*#Both are indicating that our predict variable Y "df\$Life.expectancy" is not normally distributed*

*#Shapiro-Wilk Test*

```
shapiro.test(df$Life.expectancy)
```

```
##
## Shapiro-Wilk normality test
##
## data: df$Life.expectancy
## W = 0.95676, p-value < 2.2e-16
```

*#Finding: Since df\$Life.expectancy p-value is less than .05, indicate that our y variable is not normally*

*#Kolmogorov-Smirnov Test*

```
ks.test(df$Life.expectancy, 'pnorm')
```

```
## Warning in ks.test.default(df$Life.expectancy, "pnorm"): ties should not be
## present for the Kolmogorov-Smirnov test
```

```
##
## Asymptotic one-sample Kolmogorov-Smirnov test
##
## data: df$Life.expectancy
## D = 1, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

*#Finding: Since df\$Life.expectancy p-value is less than .05, indicate that our y variable is not normally*

LM with matching dependent variable with npreg

```
model_lm <- lm(Life.expectancy~Adult.Mortality + infant.deaths + Hepatitis.B + BMI + under.five.deaths + P
summary(model_lm)
```

```
##
## Call:
## lm(formula = Life.expectancy ~ Adult.Mortality + infant.deaths +
##      Hepatitis.B + BMI + under.five.deaths + Polio + Diphtheria +
##      HIV.AIDS + GDP + thinness..1.19.years + Income.composition.of.resources +
##      Schooling, data = df, x = TRUE, y = TRUE)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -21.0326  -2.1757  -0.1334   2.1880  18.8226
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    5.387e+01  5.251e-01 102.580 < 2e-16 ***
## Adult.Mortality -1.709e-02  7.895e-04 -21.651 < 2e-16 ***
## infant.deaths    8.914e-02  7.908e-03  11.271 < 2e-16 ***
## Hepatitis.B     -1.382e-02  3.843e-03  -3.598 0.000327 ***
## BMI             3.848e-02  4.842e-03   7.947 2.76e-15 ***
## under.five.deaths -6.677e-02  5.817e-03 -11.477 < 2e-16 ***
## Polio           2.611e-02  4.501e-03   5.801 7.33e-09 ***
## Diphtheria       3.248e-02  4.764e-03   6.819 1.12e-11 ***
## HIV.AIDS        -4.808e-01  1.692e-02 -28.410 < 2e-16 ***
## GDP             6.234e-05  6.201e-06  10.054 < 2e-16 ***
## thinness..1.19.years -8.799e-02  2.241e-02  -3.927 8.83e-05 ***
## Income.composition.of.resources 6.728e+00  6.071e-01  11.082 < 2e-16 ***
## Schooling        8.110e-01  3.900e-02  20.793 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.884 on 2765 degrees of freedom
## Multiple R-squared:  0.8284, Adjusted R-squared:  0.8277
## F-statistic: 1113 on 12 and 2765 DF, p-value: < 2.2e-16
```

2. npreg on our dataset?

Note: When have time need to rerun with “x=True, y=True”, “VIF remove”, “add status.val”, “train split”

```

library(np)

## Nonparametric Kernel Methods for Mixed Datatypes (version 0.60-16)
## [vignette("np_faq",package="np") provides answers to frequently asked questions]
## [vignette("np",package="np") an overview]
## [vignette("entropy_np",package="np") an overview of entropy-based methods]

# n <- names(df)
# f <- as.formula(paste("df$Life.expectancy ~", paste(n[!n %in% "Life.expectancy"], collapse = " + ")))
#
# model_np <- npregbw(Life.expectancy ~ Adult.Mortality + infant.deaths + Hepatitis.B + BMI + under.five.d

# model_np <- npreg(bws = model_np)
# summary(model_np)
model_np <- readRDS("model_np.rds") #PreTrained Model
summary(model_np)

##
## Regression Data: 2778 training points, in 12 variable(s)
##           Adult.Mortality infant.deaths Hepatitis.B      BMI
## Bandwidth(s):      389457535      6733757    225092161 79216285
##           under.five.deaths  Polio Diphtheria HIV.AIDS      GDP
## Bandwidth(s):      95308072 5825954    19248839 1.393258 167351562078
##           thinness..1.19.years Income.composition.of.resources Schooling
## Bandwidth(s):      37667667                                1071202 15165344
##
## Kernel Regression Estimator: Local-Linear
## Bandwidth Type: Fixed
## Residual standard error: 3.345092
## R-squared: 0.8722143
##
## Continuous Kernel Type: Second-Order Gaussian
## No. Continuous Explanatory Vars.: 12

# objects()
# find("model_np")
#
# saveRDS(model_np,"model_np.rds")

#npsigtest_npreg <- npsigtest(model_np)    #10 Hrs to run...

```

3. LASSO and Neuralnet Two different supervised algorithms tried on the dataset. They do not have the constraint of a normal distribution for response variable.

First did a train and test split so we can measure the MSE and compare how each of the models are performing interms of minimizing MSE.

```

library(glmnet)

## Loading required package: Matrix

##
## Attaching package: 'Matrix'

```

```

> npsigtest(model_np)

Kernel Regression Significance Test
Type I Test with IID Bootstrap (399 replications, Pivot = TRUE, joint = FALSE)
Explanatory variables tested for significance:
Adult.Mortality (1), infant.deaths (2), Hepatitis.B (3), BMI (4), under.five.deaths (5), Polio (6), Diphtheria (7), HIV.AIDS (8),
GDP (9), thinness..1.19.years (10), Income.composition.of.resources (11), Schooling (12)

Bandwidth(s):      Adult.Mortality infant.deaths
                  389457535      6733757
Bandwidth(s):      Hepatitis.B      BMI under.five.deaths
                  225092161 79216285      95308072
Bandwidth(s):      Polio Diphtheria HIV.AIDS
                  5825954 19248839 1.393258
Bandwidth(s):      GDP thinness..1.19.years
                  167351562078      37667667
Bandwidth(s):      Income.composition.of.resources
                  1071202
Bandwidth(s):      Schooling
                  15165344

Individual Significance Tests
P Value:
Adult.Mortality      < 2e-16 ***
infant.deaths        < 2e-16 ***
Hepatitis.B          0.047619 *
BMI                  < 2e-16 ***
under.five.deaths    < 2e-16 ***
Polio                < 2e-16 ***
Diphtheria           < 2e-16 ***
HIV.AIDS             < 2e-16 ***
GDP                  < 2e-16 ***
thinness..1.19.years < 2e-16 ***
Income.composition.of.resources < 2e-16 ***
Schooling            < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

Figure 1: npsigtest\_npreg result

```
## The following objects are masked from 'package:tidyr':
```

```
##
```

```
##      expand, pack, unpack
```

```
## Loaded glmnet 4.1-6
```

```
#70:30 split for train and test
```

```
df1<-df[,c('Life.expectancy','Adult.Mortality','infant.deaths','under.five.deaths',
           'Hepatitis.B','BMI','Polio','Diphtheria',
           'HIV.AIDS','thinness..1.19.years','Income.composition.of.resources','Schooling','GDP','Status.val')]
```

```
ind <- sample(1:nrow(df1), 2000)
```

```
traino <- df1[ind,]
```

```
testo <- df1[-ind,]
```

Linear model

```
lmmodtr <- lm(traino[,1]~.,data=traino[,-1],x=TRUE, y=TRUE)
summary(lmmodtr)
```

```
##
```

```
## Call:
```

```
## lm(formula = traino[, 1] ~ ., data = traino[, -1], x = TRUE,
```

```
##      y = TRUE)
```

```
##
```

```
## Residuals:
```

```
##      Min      1Q   Median      3Q      Max
```

```
## -21.0080 -2.1304 -0.2136 2.1919 18.4589
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)      5.403e+01  6.282e-01  86.008 < 2e-16 ***
## Adult.Mortality -1.679e-02  9.212e-04 -18.222 < 2e-16 ***
## infant.deaths    8.494e-02  9.464e-03   8.974 < 2e-16 ***
## under.five.deaths -6.384e-02  7.028e-03  -9.082 < 2e-16 ***
## Hepatitis.B      -1.514e-02  4.440e-03  -3.409 0.000664 ***
## BMI              3.374e-02  5.655e-03   5.966 2.87e-09 ***
## Polio            2.999e-02  5.446e-03   5.506 4.14e-08 ***
## Diphtheria       3.174e-02  5.810e-03   5.463 5.27e-08 ***
## HIV.AIDS         -4.879e-01  1.922e-02 -25.389 < 2e-16 ***
## thinness..1.19.years -8.150e-02  2.743e-02  -2.971 0.003001 **
## Income.composition.of.resources 7.332e+00  7.380e-01   9.934 < 2e-16 ***
## Schooling        7.483e-01  4.836e-02  15.474 < 2e-16 ***
## GDP              4.822e-05  7.642e-06   6.311 3.42e-10 ***
## Status.val       1.444e+00  2.911e-01   4.960 7.65e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.894 on 1986 degrees of freedom
## Multiple R-squared:  0.8299, Adjusted R-squared:  0.8288
## F-statistic: 745.3 on 13 and 1986 DF,  p-value: < 2.2e-16
```

Another test to see if the above parametric model specification is correct.

```
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
```

```
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
##      as.Date, as.Date.numeric
```

```
resettest(lmmodtr)
```

```
##
```

```
## RESET test
```

```
##
```

```
## data:  lmmodtr
```

```
## RESET = 106.3, df1 = 2, df2 = 1984, p-value < 2.2e-16
```

```
#LASSO
```

```
library(glmnet)
```

```
y <- traino$Life.expectancy
```

```
x <- data.matrix(traino[,-1])
```

```
#k-fold cross-validation to find optimal lambda value\
```

```
#cv default is 10 fold
```

```
cv_model <- cv.glmnet(x, y, alpha = 1)
```

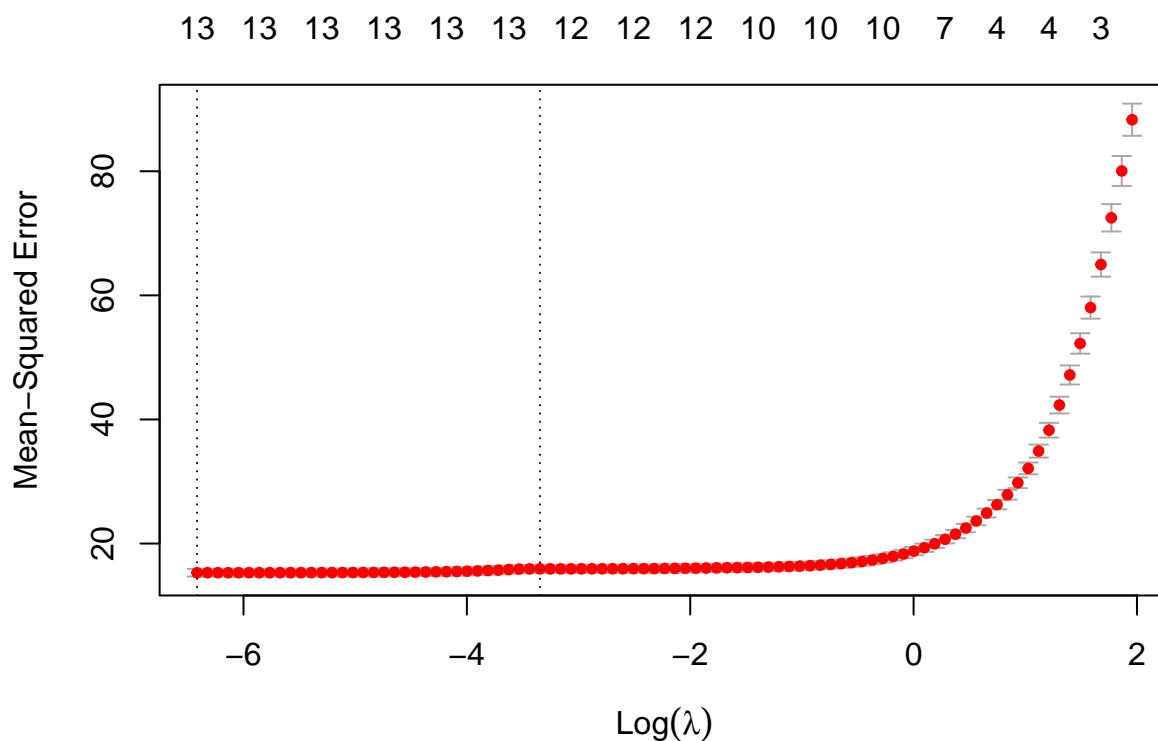
```
#optimal lambda value that minimizes test MSE
```

```
best_lambda <- cv_model$lambda.min
```

```
best_lambda
```

```
## [1] 0.00163488
```

```
plot(cv_model)
```



```
#coefficients of best model
```

```
lasmod <- glmnet(as.matrix(traino[,-1]),traino$Life.expectancy, alpha = 1, lambda = best_lambda)
```

```
coef(lasmod)
```

```
## 14 x 1 sparse Matrix of class "dgCMatrix"
```

```
## s0
```

```
## (Intercept) 5.392337e+01
```

```
## Adult.Mortality -1.684863e-02
```

```
## infant.deaths 7.639572e-02
```

```
## under.five.deaths -5.750919e-02
```

```
## Hepatitis.B -1.520809e-02
```

```
## BMI 3.377236e-02
```

```
## Polio 3.014579e-02
```

```
## Diphtheria 3.228452e-02
```

```
## HIV.AIDS -4.884856e-01
```

```
## thinness..1.19.years -7.938873e-02
```

```
## Income.composition.of.resources 7.405194e+00
```

```
## Schooling 7.491215e-01
```

```
## GDP 4.785712e-05
```

```
## Status.val 1.421325e+00
```



```

#linear model
mselm_tel <- mean((testo[,1]-predict(lmmodtr, newdata=testo))^2)

#lasso
mselas_tel <- mean((testo[,1]-predict(lasmod, newx=as.matrix(testo[,,-1]))))^2)
print(mselm_tel)

```

```
## [1] 14.40087
```

```
print(mselas_tel)
```

```
## [1] 14.44553
```

```
#MSE comparison
```

```

library(nnet)
#18 MSE
for(i in 1:62){

  set.seed(4521)

  train_lin <- nnet(traino[,1]~., data=traino, size=i, linout=TRUE, trace=FALSE)
#calculating mse

  mse_nnet_lin <- mean((testo[,1]-(predict(train_lin, newdata=testo))))^2)
  print(paste("Number of hidden layer variables:", i))
  print(paste("MSE:",mse_nnet_lin))

}

```

```

## [1] "Number of hidden layer variables: 1"
## [1] "MSE: 84.9287586652076"
## [1] "Number of hidden layer variables: 2"
## [1] "MSE: 84.9287586700756"
## [1] "Number of hidden layer variables: 3"
## [1] "MSE: 84.92875866882"
## [1] "Number of hidden layer variables: 4"
## [1] "MSE: 42.0245501405911"
## [1] "Number of hidden layer variables: 5"
## [1] "MSE: 41.6285808104543"
## [1] "Number of hidden layer variables: 6"
## [1] "MSE: 63.8293937678412"
## [1] "Number of hidden layer variables: 7"
## [1] "MSE: 37.8998418076048"
## [1] "Number of hidden layer variables: 8"
## [1] "MSE: 38.6854243100164"
## [1] "Number of hidden layer variables: 9"
## [1] "MSE: 38.5345589828876"
## [1] "Number of hidden layer variables: 10"
## [1] "MSE: 35.7775654641726"
## [1] "Number of hidden layer variables: 11"
## [1] "MSE: 65.751144029649"
## [1] "Number of hidden layer variables: 12"
## [1] "MSE: 43.145801613769"

```

```

## [1] "Number of hidden layer variables: 13"
## [1] "MSE: 34.8962614382049"
## [1] "Number of hidden layer variables: 14"
## [1] "MSE: 41.6457480144832"
## [1] "Number of hidden layer variables: 15"
## [1] "MSE: 36.4236880838662"
## [1] "Number of hidden layer variables: 16"
## [1] "MSE: 38.1882379552416"
## [1] "Number of hidden layer variables: 17"
## [1] "MSE: 36.2011179476809"
## [1] "Number of hidden layer variables: 18"
## [1] "MSE: 51.3267770383956"
## [1] "Number of hidden layer variables: 19"
## [1] "MSE: 43.0245548694463"
## [1] "Number of hidden layer variables: 20"
## [1] "MSE: 38.8123009855177"
## [1] "Number of hidden layer variables: 21"
## [1] "MSE: 39.7832121600343"
## [1] "Number of hidden layer variables: 22"
## [1] "MSE: 75.1714055070615"
## [1] "Number of hidden layer variables: 23"
## [1] "MSE: 26.1531760720715"
## [1] "Number of hidden layer variables: 24"
## [1] "MSE: 31.1927918834911"
## [1] "Number of hidden layer variables: 25"
## [1] "MSE: 28.0752553140109"
## [1] "Number of hidden layer variables: 26"
## [1] "MSE: 60.509480652386"
## [1] "Number of hidden layer variables: 27"
## [1] "MSE: 26.425434187242"
## [1] "Number of hidden layer variables: 28"
## [1] "MSE: 25.8917307608599"
## [1] "Number of hidden layer variables: 29"
## [1] "MSE: 27.2387242435489"
## [1] "Number of hidden layer variables: 30"
## [1] "MSE: 52.3143322109348"
## [1] "Number of hidden layer variables: 31"
## [1] "MSE: 19.7582018006612"
## [1] "Number of hidden layer variables: 32"
## [1] "MSE: 35.3817174696006"
## [1] "Number of hidden layer variables: 33"
## [1] "MSE: 30.5127073614838"
## [1] "Number of hidden layer variables: 34"
## [1] "MSE: 27.267443939184"
## [1] "Number of hidden layer variables: 35"
## [1] "MSE: 29.2101944475564"
## [1] "Number of hidden layer variables: 36"
## [1] "MSE: 24.4039915836126"
## [1] "Number of hidden layer variables: 37"
## [1] "MSE: 21.7648341173756"
## [1] "Number of hidden layer variables: 38"
## [1] "MSE: 22.6077227970481"
## [1] "Number of hidden layer variables: 39"
## [1] "MSE: 22.0765629170657"
## [1] "Number of hidden layer variables: 40"
## [1] "MSE: 22.2381262208709"

```

```
## [1] "Number of hidden layer variables: 41"
## [1] "MSE: 24.2682893529436"
## [1] "Number of hidden layer variables: 42"
## [1] "MSE: 19.7016863698411"
## [1] "Number of hidden layer variables: 43"
## [1] "MSE: 23.7680443355182"
## [1] "Number of hidden layer variables: 44"
## [1] "MSE: 23.4654668703972"
## [1] "Number of hidden layer variables: 45"
## [1] "MSE: 31.5953814671475"
## [1] "Number of hidden layer variables: 46"
## [1] "MSE: 28.9946399402505"
## [1] "Number of hidden layer variables: 47"
## [1] "MSE: 21.2548398865986"
## [1] "Number of hidden layer variables: 48"
## [1] "MSE: 20.4817592350449"
## [1] "Number of hidden layer variables: 49"
## [1] "MSE: 24.7596842254634"
## [1] "Number of hidden layer variables: 50"
## [1] "MSE: 23.9440512481816"
## [1] "Number of hidden layer variables: 51"
## [1] "MSE: 23.9309931478234"
## [1] "Number of hidden layer variables: 52"
## [1] "MSE: 19.6612111132269"
## [1] "Number of hidden layer variables: 53"
## [1] "MSE: 27.4823595896189"
## [1] "Number of hidden layer variables: 54"
## [1] "MSE: 20.401451756482"
## [1] "Number of hidden layer variables: 55"
## [1] "MSE: 31.2871615062158"
## [1] "Number of hidden layer variables: 56"
## [1] "MSE: 36.3112742346627"
## [1] "Number of hidden layer variables: 57"
## [1] "MSE: 17.0058732491885"
## [1] "Number of hidden layer variables: 58"
## [1] "MSE: 28.5119369229906"
## [1] "Number of hidden layer variables: 59"
## [1] "MSE: 23.2298831423284"
## [1] "Number of hidden layer variables: 60"
## [1] "MSE: 21.3161468806729"
## [1] "Number of hidden layer variables: 61"
## [1] "MSE: 30.3555712125907"
## [1] "Number of hidden layer variables: 62"
## [1] "MSE: 19.5595739572269"
```

##Pending 1. np specification test

```
# X <- data.frame(df$Adult.Mortality,df$infant.deaths,df$Hepatitis.B,df$BMI,df$under.five.deaths,df$Polio,
#
# result_npcms <- npcms test(model=model_lm, xdat=X, ydat=df$Life.expectancy) #33Hours to run
```

```
# result_npcms
```

```
# objects()
# find("result_npcms")
#
```

```
# saveRDS(result_npcms,"result_npcms.rds")
result_npcms <- readRDS("result_npcms.rds") #PreTrained Model
summary(result_npcms)
```

```
##
## Consistent Model Specification Test
## Parametric null model: lm(formula = Life.expectancy ~ Adult.Mortality +
##                          infant.deaths + Hepatitis.B + BMI + under.five.deaths
##                          + Polio + Diphtheria + HIV.AIDS + GDP +
##                          thinness..1.19.years +
##                          Income.composition.of.resources + Schooling, data =
##                          df, x = TRUE, y = TRUE)
## Number of regressors: 12
## IID Bootstrap (399 replications)
##
## Test Statistic 'Jn': 21.17521      P Value: < 2.22e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Null of correct specification is rejected at the 0.1% level
```

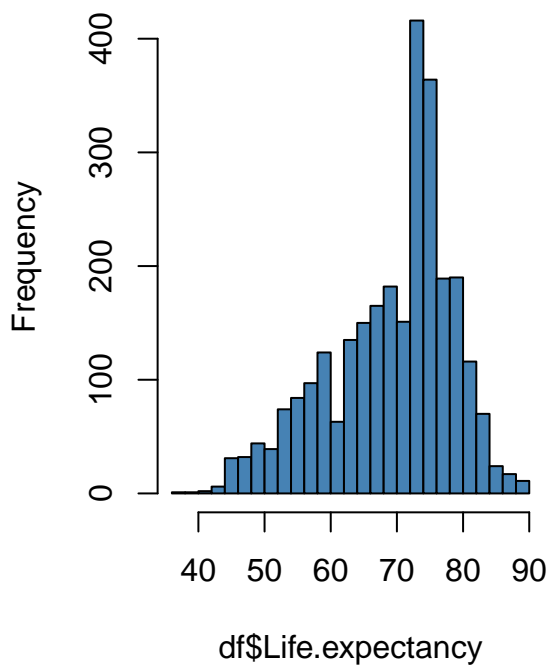
## 2. Visualizing bimodal distribution of yS

```
#Histogram & QQPlot
par(mfrow=c(1,2))
hist(df$Life.expectancy, col='steelblue', main='Life.expectancy_Histogram',breaks = 35)
#not really a good "bell-shape"
qqnorm(df$Life.expectancy, main='Life.expectancy_QQplot')
S#most of the data is not fall along a straight diagonal line
```

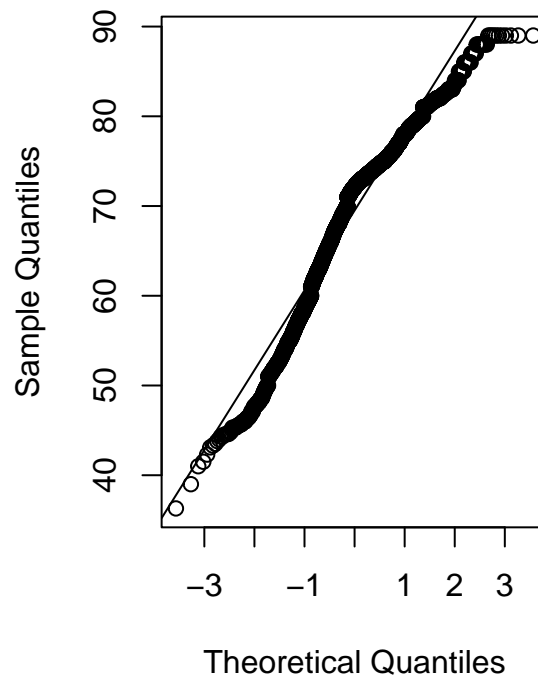
```
## function (object, brief, ...)
## {
##   UseMethod("S")
## }
## <bytecode: 0x0000019139fb3508>
## <environment: namespace:car>
```

```
qqline(df$Life.expectancy)
```

Life.expectancy\_Histogram



Life.expectancy\_QQplot



*#Both are indicating that our predict variable Y "df\$Life.expectancy" is not normally distributed*